

PERSISTENT HOMOLOGY WITH APPLICATION TO A CONNECTIVITY-LOSS FUNCTION FOR AUTOENCODERS

JUNFEI SUN

ABSTRACT. This paper discusses the persistent homology starting from simplicial homology. We then turn to the efficient calculation method of persistence regarding persistent homology using matrix reduction and pairing of positive and negative simplices. We eventually establish a structural loss function for autoencoders to carry out one-class learning tasks using persistent homology.

CONTENTS

1. Introduction	1
2. Simplicial homology	2
2.1. Simplicial Complex	2
2.2. Construction of Homology	3
3. Persistent homology	6
3.1. Persistence	7
3.2. Pairing and Matrix Reduction	8
4. Connectivity Loss with Persistent Homology	13
4.1. Autoencoder	13
4.2. Connectivity Loss Function	14
Acknowledgments	22
References	23

1. INTRODUCTION

This paper begins by introducing simplicial homology and simplicial mapping in [Section 2.1](#), leading to the construction of simplicial homology and properties of chain groups and boundary groups in [Section 2.2](#). We also discuss calculating the Betti number, specifically for 0-dimensional holes, in [Proposition 2.25](#) and [Proposition 2.26](#).

Before moving to [Section 3](#), we explain induced mapping between homology groups from simplicial mapping. In [Section 3](#), we use induced mapping and the concept of filtration to construct persistent homology groups. We explore the birth and death of homology groups to understand persistence in [Section 3.2](#), focusing on an efficient method to calculate it using matrix reduction and pairing of positive and negative simplices.

After [Section 2](#) and [Section 3](#) cover simplicial and persistent homology, the paper explores a structural loss function for autoencoders in [Section 4](#). This loss function

is based on persistent homology and aims to reshape the latent space of autoencoders efficiently for one-class learning tasks. We also discuss the differentiability of the connectivity loss function and its impact on training results. Finally, [Section 4.2.4](#) briefly explains how these structural features provide a scoring function for one-class learning.

2. SIMPLICIAL HOMOLOGY

2.1. Simplicial Complex.

2.1.1. *Construction of simplicial complex.* We start our discussion of simplicial homology with the construction of the notion of a simplicial complex which will play an essential role in the construction of homology later. We first give definitions related to affine combinations.

Definitions 2.1. Let $u = \{u_0, u_1, u_2, \dots, u_k\}$ be a finite set of points in \mathbb{R}^d . Let $x = \sum_{i=0}^k \lambda_i u_i$ be an affine combination. We call x a **convex combination** if λ_i is non-negative for any $0 \leq i \leq k$. The set of all points that are convex combinations of u is called **convex hull**. We call the convex hull of $k + 1$ affinely independent points a **k-simplex** (k is defined to be the **dimension** of this simplex) and denote it by $\sigma = \text{conv}\{u_0, u_1, u_2, \dots, u_k\}$.

Any subset of affinely independent points is still affinely independent. This means that any subset of affinely independent points has its corresponding convex hull and can define a simplex.

Definitions 2.2. Let $u = \{u_0, u_1, u_2, \dots, u_k\}$ be a finite set of points in \mathbb{R}^d . Let $\sigma = \text{conv}\{u_0, u_1, u_2, \dots, u_k\}$. We call the convex hull of a non-empty subset of u a **face** τ of σ and write $\tau \leq \sigma$. We call the face **proper** if it's the convex hull of a proper subset of u and write $\tau < \sigma$. For any τ being a (proper) face of σ , we call σ a (proper) **coface** of τ . The **boundary** of σ , denoted by $bd\sigma$, is the union of all proper faces of σ . The **interior** of σ , denoted by $\text{int}\sigma$ is everything that is contained in σ but not the boundary of σ , i.e. $\text{int}\sigma = \sigma / bd\sigma$

Proposition 2.3. *Any $x \in \sigma$ belongs to the interior of one and only one face of σ*

Definitions 2.4. A **simplicial complex** K is a collection of simplices that satisfies the following conditions. 1) If a simplex σ is contained in K , then all its faces are also contained in K . 2) For any two simplices in K , the intersection is either a face of both simplices or empty. The **dimension** of simplicial complex K is the maximum dimension of the simplices contained in K . The **underlying space** of K , denoted by $|K|$, is a topological space (X, τ) with X being the union of all simplices contained in K and τ being the topology defined by the Euclidean space in which those simplices lie.

2.1.2. *Simplicial Mapping.* Now, we can construct a type of mapping between simplices that are eventually going to resemble continuous maps between topological spaces. Simplicial mapping would serve as an ingredient for persistent homology which we would introduce in [Section 3](#). Now, we will start by introducing a mapping between vertices of different simplices from which a simplicial map induces.

Definitions 2.5. A **vertex map** is a map $\phi : \text{Vert}K \rightarrow \text{Vert}L$ with K and L being simplicial complexes, i.e. it maps with the vertices of every simplex in K

map to vertices of a simplex in L . A **simplicial map** can be induced from every vertex map ϕ , which is a function $f : |K| \rightarrow |L|$ with

$$f(x) = \sum_{i=0}^k \beta_i(x) \phi(u_i)$$

where $\beta_i(x)$ is determined as the following. Let K have vertices $u = \{u_0, u_1, u_2, \dots, u_k\}$. Then for any point x contained in K , we know that x belongs to the interior of exactly one simplex by [Proposition 2.3](#). Let that simplex be $\sigma = \{u_{j_1}, u_{j_2}, u_{j_3}, \dots, u_{j_q}\}$ where $j_a < j_b$ for any $a < b$ and $j_q \leq k$. We can then represent x with $x = \sum_{i=0}^q \lambda_i u_i$ in which all λ_i are positive. Let $\beta_i(x) = \lambda_i$ if $i \in \{j_1, j_2, \dots, j_q\}$ and $\beta_i(x) = 0$ for $i \notin \{j_1, j_2, \dots, j_q\}$.

It's worth mentioning a key property of the simplicial map before we move on to the construction of homology which is that it preserves simplices across complexes. To put it formally, we have the following proposition.

Proposition 2.6. *Consider a simplicial map $f : |K| \rightarrow |L|$. Its image of any simplex in complex K is a simplex in complex L (not necessarily of the same dimension).*

Remark 2.7. Now that we know this function preserves simplices and is linear in each simplex. We can drop the underlying space of K and L and just denote $f : K \rightarrow L$

2.2. Construction of Homology. With the construction of a simplicial complex, we can now start constructing the notion of homology groups. Generally, homology groups capture information of meaningful topological information on spaces for discerning and counting 'n-dimensional holes' for topological spaces. Intuitively, an "n-dimensional hole" is an n-dimensional subcomplex that is not a boundary of a (n+1)-dimensional subcomplex. We will return to this intuition after constructing the language of homology groups.

Definition 2.8. For a simplicial complex K and a dimension p , a **p-chain** is the formal sum of all p-simplices σ_i in K . The formal sum takes the form of

$$c = \sum a_i \sigma_i$$

in which a_i denotes coefficients assigned to each p-simplex. Additionally, we define the addition between chains component-wise. In other words, for any $c = \sum a_i \sigma_i$ and $c' = \sum b_i \sigma_i$, we define $c + c' = \sum (a_i + b_i) \sigma_i$

Remark 2.9. We will assume a_i to be *mod2* coefficients, which means that they can be either 0 or 1 with the following computational rules: $0+0 = 0, 0+1 = 1, 1+1 = 0$ for computational topology purposes. With such restrictions, we can intuitively consider σ_i with a coefficient of 1 to mean "includes σ_i " and 0 to mean the opposite.

Proposition 2.10. *The set of all p-chains in a simplicial complex together with addition forms an abelian group of p-chains C_p .*

Remark 2.11. Note that we can define C_p for any integer p . For p smaller than 0 or bigger than the dimension of the simplicial complex, C_p is a group with only 0.

With *mod2* coefficients, we can consider p-chains simply as a collection of p-simplices. Then it makes sense to talk about the boundary of that collection.

Definition 2.12. For any p-simplex $\sigma = \text{conv}\{u_0, u_1, \dots, u_q\}$, we take the **boundary** of that p-simplex to be the formal sum of all (p-1)-simplices in the complex that are faces of this p-simplex. We write

$$\partial_p \sigma = \sum_{i=0}^q \text{conv}\{u_0, \dots, \hat{u}_i, \dots, u_p\}$$

Then for a p-chain $c = \sum a_i \sigma_i$, we define the **boundary** of this p-chain to be the formal sum of boundaries of p-simplices in that chain. We write

$$\partial_p c = \sum a_i \partial_p \sigma_i$$

Remark 2.13. We can see that the boundary ∂_p then constitutes a map from p-chains to (p-1)-chains. We can then write $\partial_p : C_p \rightarrow C_{p-1}$

It should be clear from the definition that ∂_p is a homomorphism as it respects group operations. Formally, for c and c' being p-chains $\partial_p(c + c') = \partial_p c + \partial_p c'$

Definition 2.14. A **chain complex** is a sequence of chain groups connected by boundary maps.

We now take a look at two subgroups of C_p that are particularly useful for the construction of homology groups.

Definitions 2.15. A **p-cycle** is a p-chain whose boundary is 0. A **p-boundary** is a p-chain that is the boundary of a (p+1)-chain.

Proposition 2.16. *In a simplicial complex K , the set of all p-cycles forms a subgroup of the group of p-chains, denoted by $\mathbf{Z}_p = \mathbf{Z}_p(\mathbf{K})$. The same is true for the set of all p-boundaries which is denoted by $\mathbf{B}_p = \mathbf{B}_p(\mathbf{K})$.*

Remark 2.17. We will from now on denote the group of p-cycles using Z_p and the group of p-boundaries using B_p

Lemma 2.18. *For any integer p , $\partial_p \partial_{p+1} d = 0$ for any (p+1)-chain d .*

Proof. We will show $\partial_p \partial_{p+1} \tau = 0$ for any (p+1)-simplex τ . The statement follows immediately. Note that $\partial_{p+1} \tau$ consists of all the p-faces of τ .

Now, we will first show that every (p-1)-face of τ is a face of exactly two p-faces of τ . Every (p-1)-face of τ is the simplex defined by p vertices of τ , let this set of vertices be $\{u_0, u_1, \dots, u_{p-1}\}$. This leaves only two vertices of τ which we denote u_p and u_{p+1} not included in this (p-1)-face. Then the only two p-simplex that can have this (p-1)-face as a face are $\{u_0, u_1, \dots, u_{p-1}, u_p\}$ and $\{u_0, u_1, \dots, u_{p-1}, u_{p+1}\}$ which is what we want to show.

We note that $\partial_p \partial_{p+1} \tau$ as this the formal sum of all (p-1)-faces of all p-faces of τ , includes all (p-1)-faces of τ and exactly twice for each (p-1)-face. This proves that all coefficients of all terms eventually get zeroed out and proves the claim. \square

Corollary 2.19. *For any integer p , we have B_p is a subgroup of Z_p*

Proof. [Lemma 2.18](#) proves that every boundary of a p-chain is a p-cycle. This means that any $c \in B_p$ also satisfies $c \in Z_p$ which proves the corollary. \square

The fact that B_p is a subgroup of Z_p allows us to talk about the quotient groups formed by Z_p/B_p

Definitions 2.20. We call the p-th cycle group module the p-th boundary group the **p-th homology group**, in notation, $H_p = Z_p/B_p$. The p-th Betti Number is defined to be the rank of the p-th homology group, in notations, $\beta_p = \text{rank}(H_p)$. A class in H_p is called a **homology class**

By the definition of taking quotient groups, any cycle c in Z_p can give rise to a homology class in B_p denoted $[c]$ such that $[c] = \{c + b : b \in B_p\}$. Additionally, every cycle in Z_p is contained in one and only one homology class.

If we define the addition of the two classes to be $[c] + [c'] = [c + c']$, we have an addition operation between classes that is independent of the representatives chosen and consequently well-defined. It's easy to check that the homology classes together with this addition operation indeed form a group H_p .

Definition 2.21. If c and c' belong to the same homology class, they are said to be **homologous**, denoted by $c \sim c'$

Notice that for any c and c' to be homologous, they differ by a boundary, i.e. $(c - c') \in B_p$. This along with the fact that every cycle in Z_p is contained in one and only one homology class gives us an intuitive understanding of H_p : It's a partition of Z_p into classes of cycles that only differ by a boundary. The extraction of this topological feature helps us detect p-dimension holes. It turns out that they can be counted by the p-th Betti number (except 0-dimension holes).

We will now explore how to calculate Betti numbers and how it's linked to the number of holes. To do that, we need to first examine some properties of quotient groups.

Definition 2.22. Let V be a vector space over a field \mathbb{K} and N a subspace of V . An equivalence relation \sim_N on V is defined to be that $x \sim_N y$ if $x - y \in N$. We define the **quotient space** V/N to be the set of V/\sim_N which is the set of all equivalence classes of V formed by \sim , along with addition being $[v] + [u] = [v + u]$ and scalar multiplication $k[v] = [kv]$ for any $v, u \in V$ and $k \in \mathbb{K}$

Remark 2.23. It's easy to check that V/N does form a vector space

Proposition 2.24. For any vector space V , if V is of finite dimension and N is its subspace, we have $\dim V/N = \dim V - \dim N$

[Proposition 2.24](#) provides us with a concrete way to calculate Betti Numbers.

Proposition 2.25. For any integer p , $\beta_p = \text{rank} Z_p - \text{rank} B_p$

Proof. Firstly, Let n be the number of p-simplices in the simplicial complex. As C_p is a group of cycles with modulo 2 coefficients, we can see that C_p is isomorphic to \mathbb{Z}_2^n which is an n-dimensional vector space obtained from vectors with modulo 2 coefficients through $\phi : C_p \rightarrow \mathbb{Z}_2^n$, $\phi(\sum_{i=1}^n \lambda_i \sigma_i) = (\lambda_1, \lambda_2, \dots, \lambda_n)$.

Therefore, Group Z_p can be considered a subspace of \mathbb{Z}_2^n and B_p a subspace of Z_p . It then makes sense to talk about the quotient space of Z_p/B_p . From [Proposition 2.24](#), we have $\dim Z_p/B_p = \dim Z_p - \dim B_p$. Therefore, $\beta_p = \text{rank} H_p = \dim Z_p/B_p = \dim Z_p - \dim B_p = \text{rank} Z_p - \text{rank} B_p$ \square

Now that we have a way to calculate the Betti number, we still lack an explanation of how it links to the measurement of holes. Intuitively, the rank of the homology groups describes the number of "basic" p-cycles that are not bounding a (p+1)-simplex of the "cycles modulo boundaries" structure. This intuition is put in a more formal language specifically for 0-cycles in the following proposition.

Proposition 2.26. Let K be a finite simplicial complex. Then Betti number β_0 of K is the number of connected components of K

0-dimensional holes refer to edges that are required to connect the simplicial complex into a path-connected complex. This means the number of connected components minus 1 is the number of "basic" 0-dimensional holes in a simplicial complex.

Remark 2.27. This difference of 1 is specifically a problem with 0-dimensional holes. If we want β_0 to fit completely with the number of 0-dimensional holes, we can adopt reduced homology.

We close this section with a discussion of the induced map between homology groups. This induced map between homology groups will be essential to how persistent homology links homology groups together which will be discussed in the next section. Recall that for any two simplicial complexes K and L , a simplicial map $f : K \rightarrow L$ maps each simplex in K linearly to L as introduced in [Section 2.1.2](#). We can then derive a map from p-chains in K to p-chains in L for each p . To do that, we first define $f_{\#} : C_p(K) \rightarrow C_p(L)$

$$f_{\#}(\sum a_i \sigma_i) = \sum a_i \mu_i$$

where $\sum a_i \sigma_i$ is a p-chain in K and $\mu_i = f(\sigma_i)$ if $f(\sigma_i)$ is still of dimension p (by [Proposition 2.6](#), f might map a simplex to a lower-dimensional simplex) and $\mu_i = 0$ if $f(\sigma_i)$ is of a lower dimension than p .

We will start from this $f_{\#}$ to induce a map between $H_p(K)$ and $H_p(L)$. We begin with the following lemma which is easily verified.

Lemma 2.28. *For ∂_K and ∂_L being p-boundary maps in the two complexes, we have $f_{\#} \circ \partial_K = \partial_L \circ f_{\#}$*

Corollary 2.29. *For $f_{\#} : C_p(K) \rightarrow C_p(L)$, we have $f_{\#}(Z_p(K)) \subseteq Z_p(L)$ and $f_{\#}(B_p(K)) \subseteq B_p(L)$*

In words, the map $f_{\#}$ maps cycles to cycles and boundaries to boundaries. Therefore, it defines a map on the quotients which is an induced map on homology $f_* : H_p(K) \rightarrow H_p(L)$. This can be achieved by defining

$$f_*([c]) = [f_{\#}(c)], [f_{\#}(c)] = f_{\#}(c) + B_p(L)$$

$[c]$ is a homology group that's contained in $H_p(K)$ and $[f_{\#}(c)]$ is a homology group contained in $H_p(L)$. The last thing that we need to check is that this defining of f_* is well-defined, i.e. the choosing of the representation of a homology group doesn't change what a class maps to under f_* .

To show this, it suffices to show that for any $c' = c + b$ for some $b \in B_p(K)$, we have $f_{\#}(c') \in [f_{\#}(c)]$. By the definition of $f_{\#}$, we have $f_{\#}(c') = \sum (c_i + b_i) \mu_i$ with $\mu_i = f(\sigma_i)$ if $f(\sigma_i)$ still preserve dimension p and 0 if not and c_i, b_i being coefficients of c and b for the corresponding σ_i . We then have $f_{\#}(c') = \sum c_i \mu_i + \sum b_i \mu_i = f_{\#}(c) + f_{\#}(b)$. Finally, according to [Corollary 2.29](#), we have $f_{\#}(b) \in B_p(L)$ and $f_{\#}(c) \in Z_p(L)$. This shows that $f_{\#}(c') \in [f_{\#}(c)]$.

3. PERSISTENT HOMOLOGY

We've now seen that homology can detect the number of n-dimensional holes in a complex. However, when we try to derive topological features from a set of data. We need to differentiate the existence of noise. And this yields the necessity for persistent homology characterized by the dynamic variation of the homology groups when we construct the whole complex "step-by-step".

3.1. Persistence. To begin our discussion of persistent homology, we first introduce the concept of filtration.

Definition 3.1. Let K be a simplicial complex. We call a function $f : K \rightarrow \mathbb{R}$ **monotonic** if for any σ and τ that are simplices contained in K , $f(\sigma) \leq f(\tau)$ whenever σ is a face of τ .

This implies that for any $a \in \mathbb{R}$, we have $K(a) = f^{-1}(-\infty, a]$ is a subcomplex of K since it's collection of simplices in a complex and any face of a simplex contained in $K(a)$ is also contained in it.

Definition 3.2. Let $a_1 < a_2 < a_3 < \dots < a_n$ be the different values mapped to by all the simplices in K through a monotonic function $f : K \rightarrow \mathbb{R}$. Let

$$K_i = \begin{cases} K(a_i), & 1 \leq i \leq n, \\ \emptyset, & i = 0. \end{cases}$$

We define a **filtration** from f as the following increasing (in the sense of containing) sequence of complexes

$$\emptyset = K_0 \subseteq K_1 \subseteq \dots \subseteq K_n = K$$

Definition 3.3. Let A and B be two sets and $A \subset B$, the **inclusion map** from A to B is defined as $\iota : A \rightarrow B$, $\iota(x) = x$

Now, consider the inclusion map from K_i to K_j with $i < j$ as defined in [Definition 3.3](#). Notice that the inclusion map from $VertK_i$ to $VertK_j$ is a vertex map as defined in [Definitions 2.5](#). Therefore, the inclusion map from K_i and K_j can be considered as a simplicial map induced from the inclusion map from $VertK_i$ to $VertK_j$ as stated in [Definitions 2.5](#). Therefore, we can derive a map from the homology group of K_i to that of K_j from this inclusion map for any dimension p as mentioned at the end of the last section. We denote this function by $f_p^{i,j} : H_p(K_i) \rightarrow H_p(K_j)$ for each dimension p .

Therefore, the filtration naturally induces a sequence of homology groups by this map for any dimension p .

$$0 = H_p(K_0) \xrightarrow{f_p^{0,1}} H_p(K_1) \xrightarrow{f_p^{1,2}} \dots \xrightarrow{f_p^{n-1,n}} H_p(K_n) = H_p(K)$$

Definitions 3.4. We define the images of the above map induced by inclusion map **p-th persistent homology groups**. In notations, $H_p^{i,j} = Im(f_p^{i,j})$ for any $0 \leq i \leq j \leq n$. The **Betti number**, $\beta_p^{i,j}$, is still defined as the rank of this group.

We now consider its relationship between the homology group $Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$ which is well defined as $(B_p(K_j) \cap Z_p(K_i))$ is the intersection of two subgroups, thus also a group. This quotient group intuitively depicts how homology classes of K_i changed with the change of the boundary group when we move from i to j , thus containing the information on whether a homology class is still "alive" or has become trivial.

Proposition 3.5. For any integer p and any $0 \leq i \leq j \leq n$, the p -th persistence homology group $H_p^{i,j}$ is isomorphic to $Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$.

Proof. Let $\phi : H_p^{i,j} \rightarrow Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$, $\phi([c]) = [c]$ with $c \in Z_p(K_i)$. Note that in $\phi([c]) = [c]$, $[c]$ as an independent variable refers to a homology class belonging to $H_p^{i,j}$. It makes sense as the induced map between homology groups

$f_p^{i,j}$ gives $f_p^{i,j}([c]) = [\iota^{i,j}(c)] = [c]$ where $\iota^{i,j}(c)$ is the inclusion map from K_i to K_j which is a simplicial map as we've stated. Therefore, as long as $c \in Z_p(K_i)$, we have $[c] \in \text{Im}(f_p^{i,j}) = H_p^{i,j}$. Consequently, it's reasonable to talk about $[c]$ under this context. To avoid confusion, we use the notation $[c_p]$ for $[c]$ in $H_p^{i,j}$, and $[c_z]$ for $[c]$ in $Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$.

I will now show that ϕ is an isomorphism. It's easy to see that ϕ is surjective as every $c \in Z_p(K_i)$ is also contained in $Z_p(K_j)$ as mentioned above, thus there exists $[c_p]$ in $H_p^{i,j}$ such that $\phi([c_p]) = [c_z]$.

Now we show that ϕ is injective. For any $[c_z] = [c'_z]$ in $Z_p(K_i)/(B_p(K_j) \cap Z_p(K_i))$. It's easy to see that $[c_z]$ is contained in $[c_p]$ for any c . It then follows that $[c_p]$ and $[c'_p]$ are not disjoint, which indicates $[c_p] = [c'_p]$. This proves the injectivity.

Finally, the group operation is obviously preserved through ϕ as $\phi([(c + c')_p]) = [c_z] + [c'_z] = \phi([c_p]) + \phi([c'_p])$. \square

Definition 3.6. Let γ be a class in $H_p(K_i)$ with $0 \leq i \leq n$. We say that γ is **born at** K_i if it's not contained in $H_p^{i-1,i}$.

Let ω be a non-trivial homology class born at K_i , we say that it **dies at** K_j with $i \leq j \leq n$ if it merges into the boundary class as we go from K_{j-1} to K_j , i.e. $f_p^{j-1,j}(\omega) = 0$ where 0 denotes the boundary class and $\omega \neq 0$. We say that ω **doesn't die** if $f_p^{n-1,n}(\omega) \neq 0$

To close this subsection, we define the concept of persistence used to depict 'how long' a class has survived throughout the filtration. This information is important when we want to monitor the dynamic behavior of different homology classes.

Definition 3.7. Let γ be a homology class born at K_i and dies at K_j . We then define the **persistence** of γ to be the difference in the monotonic function values which induces K_i and K_j , i.e. $\text{pers}(\gamma) = a_j - a_i$. We also define the **index persistence** of γ to be $\text{pers}_{Id}(\gamma) = j - i$ and the **persistence tuple** of γ to be (i, j) . If γ doesn't die, the persistence tuple of γ is (i, ∞) .

3.2. Pairing and Matrix Reduction. We will explore an efficient way to calculate persistence in this subsection. To do so, we need to be able to track how births and deaths occur for different classes throughout a filtration. Notice that each time we move from K_i to K_{i+1} , we add simplices into K_i to obtain K_{i+1} because all the simplices in K_i are contained in K_{i+1} . Therefore, the effect on homology groups as we move from K_i to K_{i+1} will be the same as the effect of us adding those simplices one by one. Thus, we begin by considering the effect of adding any simplex into a complex during a filtration process.

Now, consider σ which is a p -simplex as defined in [Definitions 2.1](#). When we add σ into a complex K , σ can only affect p -chain groups as it's of dimension p . I claim that there are only 2 possible cases:

1) The addition of σ forms (at least) a new p -cycle:

If this happens, we call σ **positive**. We then have the following proposition.

Proposition 3.8. *A new p -cycle c created by a positive p -simplex σ added to complex K belongs to a new homology class that isn't contained in $H_p(K)$, i.e. a newly-born homology class.*

Proof. As any simplex contained in K maps to a value smaller than or equal to $f(\sigma)$ through f with f being the monotonic function that induced the filtration,

there cannot be any $(p+1)$ -simplex that takes σ as a face in K . Assume for the sake of contradiction that $[c] \in H_p(K)$. It follows that there exists a p -cycle c' in K such that $c - c' \in B_p(K)$. However, note that σ is not contained in any p -cycle c' in K . And σ is contained in $[c]$ by definition. Therefore σ must be contained in $c - c'$, thus there exists a $(p+1)$ -chain which contains σ in its boundary. It makes a contradiction against that there cannot be any $(p+1)$ -simplex that takes σ as a face in K . \square

Be aware that the addition of a positive σ might not only create one new p -chain. This will lead to multiple new homology classes. However, the next proposition will show that the number of "independent cycles" will only increase by one.

Proposition 3.9. *The addition of a positive p -simplex σ to a complex K increases the rank of Z_p by 1*

Proof. The addition of σ to a complex K to form a new complex will create a collection of newly-formed p -cycles that contain σ , denote them by $\{c_1, c_2, \dots, c_m\}$. Let $\{z_1, z_2, \dots, z_n\}$ be the original basis of Z_p of the complex K before the addition of σ to complex K . We can pick any c_i and $\{z_1, z_2, \dots, z_n, c_i\}$ forms a new basis. This is because, for any c_j , we have $c_j + c_i$ as a cycle that doesn't contain σ . This indicates $c_j + c_i$ was a cycle before the addition of σ to complex K , thus can be uniquely generated by $\{z_1, z_2, \dots, z_n\}$. Therefore, c_j can be uniquely generated by $\{z_1, z_2, \dots, z_n\}$ together with c_i which is the new basis. The case for any cycle that is not in $\{c_1, c_2, \dots, c_m\}$ is trivial. \square

Therefore, we can conclude that the addition of any positive p -simplex leads to an increase of $rank(Z_p)$ by 1 and no other changes, thus an increase of β_p by 1 according to [Proposition 2.25](#).

2) The addition of σ doesn't create a new p -cycle:

This event grants σ to be called **negative**. Under such circumstances, we first derive the following proposition

Proposition 3.10. *If σ is negative, the boundary of σ as a $(p-1)$ -cycle in K is not a $(p-1)$ -boundary before the addition of σ .*

Proof. If $\partial\sigma$ is a $(p-1)$ -boundary before the addition of σ , then there exists a p -chain d such that $\partial d = \partial\sigma$. However, this means that after the addition of σ , $\partial(d + \sigma) = \partial\sigma + \partial\sigma = 0$. This tells us that $d + \sigma$ is a new p -cycle which provides a contradiction to the hypothesis. \square

Intuitively, [Proposition 3.10](#) indicates that the addition of a negative simplex fills in a non-trivial $(p-1)$ -cycle, thus inducing the death of some $(p-1)$ -homology class. Similarly to case one, although the addition of one negative simplex may fill multiple non-trivial $(p-1)$ -cycles. Only one "independent class" will die of the addition of a negative simplex.

Proposition 3.11. *The addition of a negative p -simplex σ to a complex K increases the rank of B_{p-1} by 1*

The proof of the above proposition is similar to that of [Proposition 3.9](#) and will not be shown. And according to [Proposition 2.25](#), as $rank(Z_{p-1})$ won't change after the addition of a negative σ to complex K , we have β_{p-1} will decrease by 1

given that $\beta_{p-1} = \text{rank}(Z_{p-1}) - \text{rank}(B_{p-1})$.

Now, [Proposition 3.8](#) to [Proposition 3.11](#) provide us with a clear idea regarding how adding one simplex to a complex affects the rank of Z_p or B_{p-1} and the births and deaths of homology groups. One step from K_i to K_{i+1} is the same as several steps of adding simplices in K_{i+1} but not in K_i one at a time. This enables us to track β_p for each step of going through any filtration. Specifically, at each step of the filtration, we have

$$\beta_p = \#pos_p - \#neg_{p+1}$$

in which $\#pos_p$ denotes the number of positive p-simplices and $\#neg_{p+1}$ denotes the number of negative p-simplices. This is justified by the fact that $\beta_p = 0$ for any filtration at K_0 and any integer p .

Therefore, for any particular homology class, we can determine the positive simplex that gave birth to this homology class and the negative simplex that led to the death of this homology class. By pairing this pair of simplices, we can calculate the persistence of this homology class.

To carry out this task efficiently, we introduce a matrix representation for all simplices in a filtration that stores boundary information.

Definition 3.12. Given a sequence $[\sigma_i]$ of all simplices in a filtration induced by a monotonic function f such that 1) $f(\sigma_i) < f(\sigma_j) \Rightarrow i < j$ and 2) σ_i is a face of $\sigma_j \Rightarrow i < j$, we define the **boundary matrix** ∂ set up by this sequence with the following expression

$$\partial[i, j] = \begin{cases} 1, & \sigma_i \text{ is a codimension one face of } \sigma_j \\ 0, & \text{otherwise.} \end{cases}$$

Remark 3.13. Note that such a sequence can always be constructed for any filtration as f is monotonic.

Definitions 3.14. We define **low(j)** to be the index of the row of lowest '1' in column j . If the column contains no 1, **low(j)** is simply not defined. We also call a matrix R to be **reduced** if for any $j_1 \neq j_2$, we have $\text{low}(j_1) \neq \text{low}(j_2)$

We now introduce a reduction algorithm that possesses the ability to turn a boundary matrix ∂ with m columns into a reduced matrix:

Algorithm 1 Reduction algorithm

```

R = ∂
for j ← 1 to m do
  for j0 ← j to 1 do
    if low(j0) = low(j) then add column j0 to column j
  end if
end for
end for

```

The correctness of this algorithm is straightforward: each time we add column j_0 to j , we guarantee that $\text{low}(j) \neq \text{low}(j_0)$ because $\text{low}(j)$ decreases after this addition. Moreover, the following additions that take place in this inner for-loop won't make $\text{low}(j) = \text{low}(j_0)$ again because any addition will only decrease $\text{low}(j)$.

Therefore, after the inner for-loop for the j^{th} column ends, the j^{th} column won't have the same row index of lowest 1 with any previous columns. Inductively, the whole matrix will have no two equal row indices for the lowest 1 after the entire algorithm terminates.

Note that the reduced matrix of a given boundary matrix is not unique. We can continue adding columns after we reach a reduced matrix and this will result in different reduced matrices. However, the uniqueness of the lowest 1's is guaranteed.

Proposition 3.15. *For a given boundary matrix ∂ , its lowest 1's are unique regardless of its reduced matrix R*

Proof. For any entry R_{ij} of matrix R , we can consider its lower left submatrix of R with R_{ij} being its top, right corner entry. We can denote this submatrix R_i^j . The addition of columns doesn't change the rank of any submatrix. Therefore, we have that $\text{rank}(R_i^j) = \text{rank}(\partial_i^j)$.

We now consider the following expression:

$$r_R(i, j) = \text{rank}(R_i^j) - \text{rank}(R_{i+1}^j) + \text{rank}(R_{i+1}^{j-1}) - \text{rank}(R_i^{j-1})$$

Firstly, note that $r_R(i, j) = r_\partial(i, j)$. Secondly, we observe that any linear combination of the non-zero columns in R_i^j is non-zero because they have different row indices for the lowest 1's. It follows that $\text{rank}(R_i^j) = \text{number of non-zero columns}$. With these two observations, we move to evaluate this expression.

Case 1): If R_{ij} is a lowest 1 of the reduced matrix R , then the number of non-zero columns for R_{i+1}^j , R_{i+1}^{j-1} , and R_i^{j-1} are all exactly 1 less than that of R_i^j . Therefore, $r_R(i, j) = 1$

Case 2): If R_{ij} is not a lowest 1 of the reduced matrix R , then we can consider 2 subcases:

2a): None of the 1 to $(j-1)^{\text{th}}$ columns has a lowest 1 at row i . Under this case, R_i^j has the same number of non-zero columns with R_{i+1}^j . Similarly, R_{i+1}^{j-1} has the same number of non-zero columns with R_i^{j-1} . This gives us $r_R(i, j) = 0$.

2b): One of the 1 to $(j-1)^{\text{th}}$ columns has the lowest 1 at row i . It follows that R_i^j has one more non-zero column than R_{i+1}^j and R_i^{j-1} has one more non-zero column than R_{i+1}^{j-1} . This also gives us $r_R(i, j) = 0$.

Therefore, as long as R_{ij} is not the lowest 1, we have $r_R(i, j) = 0$.

All in all, we could use $r_R(i, j)$ as an indicator of whether R_{ij} is a lowest 1. To be more precise, $r_R(i, j) = R_{ij}$. However, we know that $r_R(i, j) = r_\partial(i, j)$. Therefore, whether R_{ij} is a lowest one depends solely on $r_\partial(i, j)$ and doesn't depend on the reduction process. \square

Corollary 3.16. (*Pairing Lemma*) *We have $i = \text{low}(j)$ if and only if $r_\partial(i, j) = 1$*

From the above, we know that the lowest 1's don't depend on the reduction process. We then ask what these lowest 1's represent. The answer to this question will lead us to [Theorem 3.17](#) which gives an efficient way to calculate the persistence.

We begin by considering the matrix R_{k-1} representing the matrix we have before the k^{th} iteration of the outer loop of [Algorithm 1](#). This is a matrix whose columns that represent simplices before σ_k are in the reduced form. As we enter the k^{th} iteration, it attempts to zero out the k^{th} column and there can be two outcomes: the k^{th} column can be zeroed out or not. We will see that whether the k^{th} column

can be zeroed out in the k^{th} iteration or not is directly linked to σ_k being a positive or negative simplex.

Case 1) The k^{th} column can be zeroed out.

Claim: case 1 implies that σ_k is positive. To see this, we first notice that by the construction of [Algorithm 1](#), any j_0^{th} column ($1 \leq j_0 \leq k-1$) being added to the k^{th} column requires they share the same row index for their lowest 1. I claim that this is only possible if σ_{j_0} is of the same dimension as σ_k .

To see the above, let this dimension of σ_k be p . Any possible 1 in the k^{th} column of row index i must correspond to a σ_i with dimension $p-1$. And if j_0^{th} column has a one at i^{th} row, σ_{j_0} must also have a 1-less-dimension face with dimension $p-1$, thus we have $dim(\sigma_{j_0}) = p$. After we add the j_0^{th} column to the k^{th} column, if there still exists any 1's in the k^{th} column, it must either come from a 1 in the j_0^{th} column or is originally from the k^{th} column. It follows that it must still correspond to a $(p-1)$ -dimensional simplex. Therefore, any other columns added to the k^{th} column follow the same argument above and have to correspond to a p -dimensional simplex. Inductively, all additions of columns in the k^{th} iteration must be with columns representing p -dimensional simplices.

Furthermore, if the k^{th} column corresponds to a p -dimensional simplex σ_k , the resultant k^{th} column after the addition represents a boundary of a p -chain that contains σ_k . By 'representing', we mean that for the k^{th} column after the iteration with 1's on row $\{i_1, i_2, \dots, i_m\}$, we have $\sum_{j=1}^m \sigma_{i_j}$ as a boundary. This can be shown inductively. For the base case when $k=1$, this is immediate. Now, assume that all 1 to $k-1$ columns represent boundaries after the addition of their iterations. Before the k^{th} iteration, the k^{th} column contains the boundary of σ_k . Let the columns that are going to be added to the k^{th} column be of column indices $\{j_1, j_2, \dots, j_n\}$. As shown above, all $\sigma_{j_1}, \sigma_{j_2}, \dots, \sigma_{j_n}$ are p -dimensional simplices. According to the induction hypothesis, any column with index $j_q, 1 \leq q \leq n$ has row indices of 1's corresponding to a boundary of a p -chain that contains σ_{j_q} . We denote this p -chain P_{j_q} . Therefore, adding them to the k^{th} column eventually gives rise to the boundary of $\sum_{t=1}^q P_{j_t} + \sigma_k$ which is a p -chain that contains σ_k .

All in all, if the k^{th} column got zeroed out after the k^{th} iteration. It means that $\partial(\sum_{t=1}^q P_{j_t} + \sigma_k) = 0$, i.e., $\sum_{t=1}^q P_{j_t} + \sigma_k$ is a cycle. This must be a new cycle as it contains σ_k . Therefore, σ_k is positive.

Case 2) The k^{th} column cannot be zeroed out

Claim: case 2 implies that σ_k is negative. Let $l = Low(j)$ after the k^{th} iteration. Also, recall that the k^{th} column represents a boundary of a p -chain after the k^{th} iteration, which is a $(p-1)$ -cycle. We denote this cycle by γ . Then we have the following three observations.

Obs1 $\partial\sigma_k$ is homologous to γ .

This follows immediately from that γ is obtained by adding $(p-1)$ -cycles that are boundaries (represented by previous columns) to $\partial\sigma_k$ (represented by the k^{th} column) when we add the previous columns to the k^{th} column during the k^{th} iteration of the outer loop of [Algorithm 1](#). Therefore, $\gamma - \partial\sigma_k \in B_{p-1}$

Obs2 Let the simplicial complex which is the collection of $\{\sigma_1, \sigma_2, \dots, \sigma_{k-1}\}$ be denoted K_{k-1} , we have $\partial\sigma_k$ is not a boundary in K_{k-1} , but becomes a boundary in K_k .

To see why $\partial\sigma_k$ is not a boundary in K_{k-1} , we first notice that the k^{th} column can't be zeroed out. This implies that $\partial\sigma_k$ can't be expressed as linear

combinations of the basis of R_{k-1} , thus not by any linear combination of boundaries of $\{\sigma_1, \sigma_2, \dots, \sigma_{k-1}\}$. It follows that $\partial\sigma_k$ can't be a boundary in $K_{k-1} = \{\sigma_1, \sigma_2, \dots, \sigma_{k-1}\}$.

Obs3 The cycle γ is created when adding σ_l . Consequently, the homology class $[\gamma]$ was created when adding σ_l . And when σ_l was added, it had no coface. In other words, γ was not a boundary in K_l .

Finally, according to (1) and (2), we know that γ was not a boundary in K_{k-1} because $\partial\sigma_k$ which is homologous to γ was not a boundary in K_{k-1} . Therefore, we have that γ dies at the time when σ_k is added. In other words, σ_k kills a homology class that is created by σ_l where $l = \text{low}(k)$. This gives us the following theorem.

Theorem 3.17. *For a reduction matrix R of a boundary matrix, let a non-zero column with index k that corresponds to σ_k of dimension p and has 1 at row $\{i_1, i_2, \dots, i_m\}$ in increasing order. We have that $[\sigma_{i_1} + \sigma_{i_2} + \dots + \sigma_{i_m}]$ represents a homology class created at $\text{low}(k)$ and killed at k .*

Therefore, with the above theorem, we can efficiently pair the positive simplex that gave birth to a homology class and a negative simplex that killed the class. This enables us to calculate the persistence of all classes throughout the filtration effectively. This concludes the current section and we will move on to how this notion of persistent homology along with an efficient way of calculating the persistence using pairing and matrix applies to a loss function for autoencoders.

4. CONNECTIVITY LOSS WITH PERSISTENT HOMOLOGY

In this section, we will introduce a structural-loss function using persistent homology for autoencoders that can be implemented in one-class learning tasks. To begin with, we first give a brief introduction to what autoencoders are.

4.1. Autoencoder. Autoencoders are a type of artificial neural network that carries out the task of encoding data under unsupervised learning circumstances. General autoencoders consist of three parts:

- (1) Encoder: Layers that compress data into a lower-dimensional representation
- (2) Bottleneck: contains the compressed lower-dimensional representation
- (3) Decoder: decodes the compressed representation of the data back to its original form

Given a data space X , we let $\{x_i\}$, $x_i \in X$ be a set of training data. We can denote $f : X \rightarrow Z \subset \mathbb{R}^n$ as the (non-)linear function that corresponds to the **encoder**. Similarly, we denote $g : Z \subset \mathbb{R}^n \rightarrow X$ as the (non-)linear function that corresponds to the **decoder**. The encoder and the decoder are decided by parameters θ and ϕ during the process of learning. The learning has a goal to find these two parameters that minimize the following expression

$$\sum_i L(x_i, g_\phi(f_\theta(x_i)))$$

with $L : X \times X \rightarrow \mathbb{R}$ being a reconstruction loss function that measures the deviation of the reconstruction of the input data $g_\phi(f_\theta(x_i))$ from the original input data set x_i . We want to minimize this deviation so that the autoencoder can effectively represent a set of data in lower dimensions. This can be made sure when we can construct the original data as much as possible from the representation.

When $Z \subset \mathbb{R}^n$ has a much smaller dimension than the dimension of X , we refer to $z = f_\theta(x)$ as the latent representation of x , and Z is termed the *latent space*.

We will now introduce a structural-loss function that controls the topological feature of the latent space. It assists the task of one-class learning by autoencoder together with the reconstructional loss function. One-class learning refers to learning to determine whether a data point belongs to a class of data using only training data points that belong to this class.

4.2. Connectivity Loss Function.

4.2.1. *Persistent homology for latent representations.* Given $r \geq 0$, we can consider the union of the closed balls with radius r around every $z_i = f_\theta(x_i)$ with respect to some metric on \mathbb{R}^n :

$$S_r = \bigcup_i \bar{B}(z_i, r)$$

In particular, we are curious about the number of connected components of S_r as a topological feature of the latent representations. According to [Proposition 2.26](#), homology groups capture the number of connected components. Therefore, we need a way to treat the z_i as vertices and construct complexes as r varies to capture the information on 0-dimensional homology groups at different r . This is closely linked to the persistent homology groups in [Definitions 3.4](#) and the persistence tuple in [Definition 3.7](#). We will first construct a filtration linked to the change of r .

Definition 4.1. Consider $S \subset \mathbb{R}^n$ in a metric space (\mathbb{R}^n, δ) with a finite number of elements. We denote the number of elements in S with $|S|$. Let $\mathcal{V}(S) = \{u \in P(\{1, \dots, |S|\}) : |u| = 1 \text{ or } 2\}$ where $P(\{1, \dots, |S|\})$ refers to the power set of $\{1, \dots, |S|\}$. Define

$$f_S : \mathcal{V}(S) \rightarrow \mathbb{R}, f_S(u) = \begin{cases} \frac{\delta(z_i, z_j)}{2}, & u = \{i, j\} \\ 0, & u = \{i\}. \end{cases}$$

Then, we define the **Vietoris-Rips index set** with respect to $r \geq 0$ as the following:

$$\mathcal{V}_r^{idx}(S) = f_S^{-1}((-\infty, r])$$

and we define the **Vietoris-Rips complex** with respect to $r \geq 0$ restricted to 1-skeleton as the following:

$$\mathcal{V}_r(S) = \bigcup_{u \in \mathcal{V}_r^{idx}(S)} \text{conv}(\{z_i : i \in u\})$$

Remark 4.2. Firstly, notice that $\mathcal{V}_r(S)$ contains all single vertices because they map to 0 through f_S . Moreover, it includes all pairs of vertices $\{z_i, z_j\}$ with half of their distance smaller or equal to r (equivalently, $\bar{B}(z_i, r) \cap \bar{B}(z_j, r) \neq \emptyset$), and the edges connecting z_i and z_j .

Now, we arrange all distances between pairs of z_i and z_j into an increase sequence: $(\epsilon_k)_{k=1}^M$. It follows that

$$\emptyset \subset \mathcal{V}_0(S) \subset \mathcal{V}_{\frac{\epsilon_1}{2}}(S) \subset \dots \subset \mathcal{V}_{\frac{\epsilon_M}{2}}(S)$$

can be viewed as a filtration as each element is a complex and this whole sequence is increasing in the sense of inclusion. We let $\epsilon_0 = 0$ and call this filtration the **Vietoris-Rips filtration** of S . Similarly, we call

$$\emptyset \subset \mathcal{V}_0^{idx}(S) \subset \mathcal{V}_{\frac{\epsilon_1}{2}}^{idx}(S) \subset \dots \subset \mathcal{V}_{\frac{\epsilon_M}{2}}^{idx}(S)$$

the **Vietoris-Rips index filtration** of S .

Remark 4.3. Vietoris-Rips index filtration is a sequence of collections of indices whereas Vietoris-Rips filtration is the sequence of complexes formed by the latent representations with the indices in the corresponding Vietoris-Rips index filtration

Now, we can construct 0-dimensional persistent homology with this Vietoris-Rips filtration like the one defined in [Definitions 3.4](#):

$$0 = H_0(\mathcal{V}_0(S)) \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_1}{2}}(S)) \rightarrow \dots \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_M}{2}}(S))$$

This persistent homology helps observe how different ϵ_i affect the connectivity of $\mathcal{V}_{\frac{\epsilon_i}{2}}$. Notice that we have an edge between v_i and v_j in $\mathcal{V}_r(S)$ if and only if $\overline{B}(z_i, r) \cap \overline{B}(z_j, r) \neq \emptyset$ according to [Remark 4.2](#). Therefore, this persistent homology also indicates how different $r = \frac{\epsilon_i}{2}$ affect the connectivity of S_r .

According to the definition of persistence tuple in [Definition 3.7](#), the above 0-dimensional persistent homology produces a multi-set of persistence tuples (i, j) with $i < j$ in which each (i, j) corresponds to a 0-dimensional homology class born at $\mathcal{V}_{\frac{\epsilon_i}{2}}(S)$ and dies at $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$. All tuples can be computed efficiently by the method of matrix reduction and pairing introduced in [Section 3.2](#). At the same time, (i, j) indicates a connected component of S_r that persists from $S_{\frac{\epsilon_i}{2}}$ to $S_{\frac{\epsilon_j}{2}}$.

Definition 4.4. We call the multi-set of all persistence tuples the **persistence barcode** of this 0-dimensional persistent homology

$$0 = H_0(\mathcal{V}_0(S)) \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_1}{2}}(S)) \rightarrow \dots \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_M}{2}}(S))$$

out of the filtration $\emptyset \subset \mathcal{V}_0(S) \subset \mathcal{V}_{\frac{\epsilon_1}{2}}(S) \subset \dots \subset \mathcal{V}_{\frac{\epsilon_M}{2}}(S)$ with each $\mathcal{V}_r(S)$ as defined in [Definition 4.1](#), and denote it as $\mathcal{B}(S)$.

Remark 4.5. It's a multi-set because several homology groups can die at the same $\mathcal{V}_{\frac{\epsilon_j}{2}}$

Proposition 4.6. *For any (i, j) being a persistence tuple of a homology class γ in the 0-dimensional persistent homology*

$$0 = H_0(\mathcal{V}_0(S)) \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_1}{2}}(S)) \rightarrow \dots \rightarrow H_0(\mathcal{V}_{\frac{\epsilon_M}{2}}(S))$$

, we have $i = 0$.

Proof. Note that any vertex is contained in S_0 , this means that all possible connected components appear at $\mathcal{V}_0(S)$. Recall that according to the discussion in [Section 3.2](#), all 0-dimensional homology classes must be created by a positive 0-simplex which is a vertex. Therefore, all 0-dimensional homology classes are also created at $\mathcal{V}_0(S)$, including γ . It follows that $i = 0$. \square

Also, for j in any (i, j) , we know that either $1 \leq j \leq M$ or $j = \infty$.

Proposition 4.7. *Fix $1 \leq j \leq M$, there exists $(0, j)$ as a persistence tuple if and only if there exists a pair of vertices v_1 and v_2 with $\delta(v_1, v_2) = \epsilon_j$ that were not in the same connected component in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$.*

Proof. To prove the forward direction, we assume $(0, j)$ is a persistence tuple. It follows that there exists a 0-dimensional homology class which dies at $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$. According to the discussion in [Section 3.2](#), this can only happen if there exists a negative simplex added when entering $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$. By the construction of Vietoris-Rips filtration, all the simplices that are added when entering $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$ are edges between pairs of vertices whose distances in between are ϵ_j . Let the edge between v_1 and v_2 be a negative simplex. According to [Proposition 3.10](#), this implies that $v_1 + v_2 \notin B_0(\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S))$, i.e. it's not a boundary before entering $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$. Within the proof of [Proposition 2.26](#), we've established that $v_1 + v_2 \in B_0(\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)) \iff$ (there exists a path joining v_1 and v_2 in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$). Therefore, we have that v_1 and v_2 are not path-connected in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$, thus not connected in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$.

Now we prove the other direction. Assume that there exists a pair of vertices v_1 and v_2 with $\delta(v_1, v_2) = \epsilon_j$ that were not in the same connected component in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$. This implies moving into $\mathcal{V}_{\frac{\epsilon_j}{2}}(S)$ adds an edge between v_1 and v_2 . Since v_1 and v_2 were not path-connected in $\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S)$, we use again the reasoning in the proof of [Proposition 2.26](#) and observe that $v_1 + v_2 \notin B_0(\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S))$. It follows that $[v_1 + v_2]$ wasn't trivial in $H_0(\mathcal{V}_{\frac{\epsilon_{j-1}}{2}}(S))$. But it turns trivial in $H_0(\mathcal{V}_{\frac{\epsilon_j}{2}}(S))$ as it's the boundary of the edge. This means that it dies at j , i.e. there exists a persistence tuple $(0, j)$ \square

Definition 4.8. Let $S \subset \mathbb{R}^n$ be a finite set and $(\epsilon_k)_{k=1}^M$ be the increasing sequence of distances of all pairs of points in S . Let $\mathcal{B}(S)$ be the persistence barcode of the Vietoris-Rips filtration of S . We now define

$$D(S) = \{d : (0, d) \in \mathcal{B}(S), d \neq \infty\}$$

as a **multi-set of death times**.

4.2.2. Construction of Connectivity Loss. With the above construction of Vietoris-Rips filtration and set of death times, we can now introduce the connectivity loss function which helps us learn the desired connectivity structure for a batch S of latent representations.

Definition 4.9. Let $D(S)$ be the set of death times for S . Given $\eta > 0$, we define

$$\mathcal{L}_\eta(S) = \sum_{d \in D(S)} |\eta - \epsilon_d|$$

as the **connectivity loss**

Remark 4.10. This connectivity loss punishes a deviation from a configuration in which all death times $\epsilon_d = D(S)$ are equal to η , i.e. the complex $\mathcal{V}_\eta(S)$ is connected. This ensures that we move towards a structure in which each representation of S will have at least one neighbor within a radius of η . We will discuss the utility of this structure for the one-class learning task of the autoencoder in [Section 4.2.3](#).

For the connectivity loss function defined in [Definition 4.9](#) to be put into the training process, we have to make sure it's possible to compute the partial derivative of this function with respect to its inputs which are the data points in S . This is essential for carrying out back-propagation which uses gradient descent. In the following analysis, we fix the metric of the metric space in which the latent spaces lie to be p-norm: $Z \subset (\mathbb{R}^n, \delta) = (\mathbb{R}^n, \|\cdot\|)$.

Definition 4.11. Let $S \subset \mathcal{R}^n$ and all latent representations $z_i \in S$. We define **death indicator function** $\mathbf{1}_{i,j} : S \times \dots \times S \rightarrow \mathbb{R}$

$$\mathbf{1}_{i,j}(z_1, z_2, \dots, z_{|S|}) = \begin{cases} 1, & \exists d \in D(S) : \epsilon_d = \|z_i - z_j\| \\ 0, & \text{otherwise.} \end{cases}$$

in which $(\epsilon_k)_{k=1}^M$ is an increasing sequence of all pairwise distances between vertices in S .

Theorem 4.12. Let $S \subset \mathcal{R}^n$ and we assume that all the distances between pairs of vertices are distinct. Let $(L)_\eta(S)$ be the connectivity loss function as defined in Definition 4.9. We have the following:

$$\mathcal{L}_\eta(S) = \sum_{\{i,j\} \subset \{1, \dots, |S|\}} |\eta - \|z_i - z_j\|| \cdot \mathbf{1}_{i,j}(z_1, \dots, z_{|S|})$$

where $\mathbf{1}_{i,j}$ is the death indicator function as defined in Definition 4.11

The above theorem can be easily verified by proving inequalities from both directions. Now we can move to show the differentiability of the loss function by introducing the lemma below.

Lemma 4.13. Let $S \subset \mathbb{R}^n$. Assume that all pairwise distances of vertices are distinct. For any $1 \leq q \leq |S|$ and $1 \leq v \leq n$, let

$$S' = \{z'_1, \dots, z'_n\}, z'_i = \begin{cases} z_i, & 1 \leq i \leq n, i \neq q \\ z_q + h \cdot e_v, & i = q. \end{cases}$$

where e_v is the v -th unit vector and $h \in \mathbb{R}$. It follows that there exists $\xi > 0$ such that $|h| < \xi \Rightarrow \mathbf{1}_{i,j}(S) = \mathbf{1}_{i,j}(S')$

Proof. To show that $\mathbf{1}_{i,j}(S) = \mathbf{1}_{i,j}(S')$, we notice that the death indicator function is determined by the Vietoris-Rips index filtration of S . Therefore, it suffices to prove S and S' have the same Vietoris-Rips index filtration.

Let $(\epsilon_k)_{k=1}^M$ be the increasing sequence of pairwise distances between vertices of S , and $(\epsilon'_k)_{k=1}^M$ be the same for S' . We will first show that $(\epsilon'_k)_{k=1}^M$ is strictly increasing when $|h|$ is smaller than a fixed value.

Notice that for the case $\|z'_i - z'_j\| \leq \|z_i - z_j\|$, we can only have $\|z'_i - z'_j\| < \|z_i - z_j\|$ when $z'_j = z_j + h \cdot e_v$ and $z'_i = z_i$. This gives us $\|z_i - z_j\| = \|z'_i - z'_j + h \cdot e_v\| \leq \|z'_i - z'_j\| + |h|$. It follows that when $\|z'_i - z'_j\| \leq \|z_i - z_j\|$, we have

$$\| \|z'_i - z'_j\| - \|z_i - z_j\| \| = \|z_i - z_j\| - \|z'_i - z'_j\| \leq \|z'_i - z'_j\| + |h| - \|z'_i - z'_j\| = |h|$$

Similarly, under the case $\|z'_i - z'_j\| \geq \|z_i - z_j\|$, we have $\|z'_i - z'_j\| \leq \|z_i - z_j\| + |h|$, which gives us

$$\| \|z'_i - z'_j\| - \|z_i - z_j\| \| = \|z'_i - z'_j\| - \|z_i - z_j\| \leq \|z_i - z_j\| + |h| - \|z_i - z_j\| = |h|$$

Combining the two results above, we have that $\| \|z'_i - z'_j\| - \|z_i - z_j\| \| \leq |h|$ for any i, j .

Let $\mu = \min_{1 \leq k < M} \epsilon_{k+1} - \epsilon_k$. As we've assumed that all pairwise distances are distinct, we can denote $\epsilon_k = \|z_{i_k} - z_{j_k}\|$ where z_{i_k} and z_{j_k} are two unique vertices.

For any $1 \leq k < M$, we have

$$\begin{aligned} \epsilon'_{k+1} - \epsilon'_k &= \|z'_{i_{k+1}} - z'_{j_{k+1}}\| - \|z'_{i_k} - z'_{j_k}\| \\ &\geq \|z_{i_{k+1}} - z_{j_{k+1}}\| - |h| - \|z_{i_k} - z_{j_k}\| - |h| \\ &= \epsilon_{k+1} - \epsilon_k - 2|h| \\ &\geq \mu - 2|h| \end{aligned}$$

It follows that $(\epsilon'_k)_{k=1}^M$ is strictly increasing if $|h| < \frac{\mu}{2}$.

Now, we move to show that the Vietoris-Rips index filtration $\emptyset \subset \mathcal{V}_0^{idx}(S) \subset \mathcal{V}_{\frac{\epsilon_1}{2}}(S) \subset \dots \subset \mathcal{V}_{\frac{\epsilon_M}{2}}^{idx}(S)$ is the same as $\emptyset \subset \mathcal{V}_0^{idx}(S') \subset \mathcal{V}_{\frac{\epsilon'_1}{2}}^{idx}(S') \subset \dots \subset \mathcal{V}_{\frac{\epsilon'_M}{2}}^{idx}(S')$

when $|h| < \frac{\mu}{2}$. We will show this inductively.

For the base case, we have that $\mathcal{V}_0^{idx}(S) = \mathcal{V}_0^{idx}(S') = \{\{1\}, \dots, \{|S|\}\}$.

Assume that $\mathcal{V}_{\frac{\epsilon_k}{2}}^{idx}(S) = \mathcal{V}_{\frac{\epsilon'_k}{2}}^{idx}(S')$. By the definition of f_S in [Definition 4.1](#), we have $f_S^{-1}(\frac{\epsilon_{k+1}}{2}) = \{\{i_{k+1}, j_{k+1}\}\}$. This is because $\epsilon_{k+1} = \|z_{i_{k+1}} - z_{j_{k+1}}\|$ and the pairwise distances between any two vertices in S are distinct by the assumption of this lemma. Also, we have $f_{S'}^{-1}(\frac{\epsilon'_{k+1}}{2}) = \{\{i_{k+1}, j_{k+1}\}\}$ because the pairwise distances between any two vertices in S' are distinct when $|h| < \frac{\mu}{2}$ because $\epsilon'_{k+1} = \|z'_{i_{k+1}} - z'_{j_{k+1}}\|$ and it's proven that $(\epsilon'_k)_{k=1}^M$ is strictly increasing when this is satisfied. Therefore, we have

$$\begin{aligned} \mathcal{V}_{\frac{\epsilon'_{k+1}}{2}}(S') &= \mathcal{V}_{\frac{\epsilon'_k}{2}}(S') \cup f_{S'}^{-1}(\frac{\epsilon'_{k+1}}{2}) \\ &= \mathcal{V}_{\frac{\epsilon_k}{2}}(S') \cup f_{S'}^{-1}(\frac{\epsilon'_{k+1}}{2}) \\ &= \mathcal{V}_{\frac{\epsilon_k}{2}}(S') \cup f_S^{-1}(\frac{\epsilon_{k+1}}{2}) \\ &= \mathcal{V}_{\frac{\epsilon_{k+1}}{2}}(S) \end{aligned}$$

Therefore, we've proven that by letting $\xi = \frac{\mu}{2}$, we have that when $|h| < \xi$, the Vietoris-Rips index filtration of S and S' are the same, thus $\mathbf{1}_{i,j}(S) = \mathbf{1}_{i,j}(S')$ for any i, j . \square

Theorem 4.14. *Let $S \subset \mathbb{R}^n$ and pairwise distances are distinct. For any $1 \leq q \leq |S|$ and $1 \leq v \leq n$, we have that the partial derivative of $\mathcal{L}_\eta(S)$ with respect to v -th coordinate of z_q exists:*

$$\frac{\partial \mathcal{L}_\eta(S)}{\partial (z_q)_v} = \sum_{\{i,j\} \subset \{1, \dots, |S|\}} \frac{\partial |\eta - \|z_i - z_j\||}{\partial (z_q)_v} \cdot \mathbf{1}_{i,j}(z_1, \dots, z_n)$$

Proof. According to [Lemma 4.13](#), let S' be defined as in [Lemma 4.13](#) for q and v , we have that $\mathbf{1}_{i,j}(S) = \mathbf{1}_{i,j}(S')$ when $|h| < \frac{\mu}{2}$, i.e. $\mathbf{1}_{i,j}$ is locally constant with respect to $(z_q)_v$. This implies that $\frac{\partial \mathbf{1}_{i,j}(S)}{\partial (z_q)_v} = 0$. By [Theorem 4.12](#), we have that $\mathcal{L}_\eta(S) = \sum_{\{i,j\} \subset \{1, \dots, |S|\}} |\eta - \|z_i - z_j\|| \cdot \mathbf{1}_{i,j}(S)$. Now, according to the product

rule:

$$\begin{aligned} \frac{\partial \mathcal{L}_\eta(S)}{\partial (z_q)_v} &= \sum_{\{i,j\} \subset \{1,\dots,|S|\}} \frac{\partial |\eta - \|z_i - z_j\||}{\partial (z_q)_v} \cdot \mathbf{1}_{i,j}(S) + |\eta - \|z_i - z_j\|| \cdot \frac{\partial \mathbf{1}_{i,j}(S)}{\partial (z_q)_v} \\ &= \sum_{\{i,j\} \subset \{1,\dots,|S|\}} \frac{\partial |\eta - \|z_i - z_j\||}{\partial (z_q)_v} \cdot \mathbf{1}_{i,j}(S) \end{aligned}$$

□

Therefore, we've proven that the connectivity loss function defined in [Definition 4.9](#) can be put into the training process through back-propagation given the existence of every partial derivative proven in [Theorem 4.14](#).

4.2.3. Connectivity loss impact analysis. Finally, we analyze the impact of this connectivity loss on the training result. If we merely train with the connectivity loss with batches of size $a \in \mathbb{R}$, by the connectivity loss in [Definition 4.9](#), we ideally obtain a parameterization θ of f_θ (the encoder as introduced in [Section 4.1](#)) that makes every a -sized random sample S to satisfies $\epsilon_d = \eta$ for every $d \in D(S)$ (all using the definitions in [Section 4.2](#)). However, as mentioned in [Section 4.1](#), we must train the autoencoder with both this connectivity loss and the reconstruction loss to ensure the autoencoder can decode the information back to its original form. Therefore, ideally, we have every ϵ_d to be in $[\alpha, \beta]$ such that $\eta \in [\alpha, \beta]$. To depict this situation, we introduce the following definition:

Definition 4.15. Let $S \in \mathbb{R}^n$ be a finite set and $(\epsilon_k)_{k=1}^M$ the increasing sequence of distances between vertices in S . We define S to be $\alpha - \beta$ -connected if $\min_{d \in D(S)} \epsilon_d = \alpha$ and $\max_{d \in D(S)} \epsilon_d = \beta$

Now, we need to be aware that as we trained the connectivity loss with only a -sized batches, only S with $|S| = a$ can satisfy that f_θ is $\alpha - \beta$ -connected. But realistically, the trained network will also be fed with samples with a size more than a . Therefore, we need to analyze how the trained encoder impacts the structure of the latent space when fed with samples more than size a .

To begin with, we will first analyze how the $\alpha - \beta$ -connected structure impacts the density of the latent space. Briefly, the trained neural network can ensure a certain density in the latent space that is linked to the size of the sample and the batch size a .

Lemma 4.16. Let $M \subset \mathbb{R}^n$ with $2 \leq a \leq |M|$ where a is the batch size used for training. If M satisfies that for any $S \subset M$ with $|S| = a$, we have S is $\alpha - \beta$ -connected, then it follows that for $d = |M| - a$ and any $z \in M$, we can specify $M_z \subset M$ with $|M_z| = d + 1$ such that $M_z \subset B(z, \beta) \setminus B(z, \alpha)$

Proof. For any $z \in M$, we can construct a set of points $\{z_1, \dots, z_{d+1}\}$ in M inductively with the following steps.

Firstly, we can find $S^1 \subset M$ such that $z \in S^1$ and $|S^1| = a$. It follows from the assumption that S^1 is $\alpha - \beta$ -connected, which implies that $z \in S^1$ has a neighbor at a distance within $[\alpha, \beta]$ (all other vertices lie in this range of distance from z). We denote this neighbor to be z_1 with $z_1 \in B(z, \beta) \setminus B(z, \alpha)$ and $z_1 \in S^1$. Let $M_z^1 = z_1$

Now assume that for $i \leq d = |M| - a$, we can consider $S^i \in M$ with $z_i \in S^i$ and $|S^i| = a$ which gives us $M_z^i = \{z_1, \dots, z_i\}$. We have $|M \setminus M_z^i| = |M| - i \geq |M| - d = a$.

Therefore, we can find $S^{i+1} \subset M \setminus M_z^i$ such that $z \in S^{i+1}$ and $|S^{i+1}| = a$. This implies that S^{i+1} is $\alpha - \beta$ -connected, and we can find $z_{i+1} \in B(z, \beta) \setminus B(z, \alpha)$ and $z_{i+1} \in S^{i+1}$ by the same logic as that for z_1 . Note that z_{i+1} is distinct from any z_j with $j < i$ because $z \in S^{i+1} \subset M \setminus M_z^i$, which means that we can construct $M_z^{i+1} = \{z_1, \dots, z_{i+1}\}$.

Therefore, the above proves that we can construct M_z^{i+1} with any $i \leq d = |M| - a$. Therefore, it follows that we can construct $M_z^{d+1} = \{z_1, \dots, z_{d+1}\}$ and $M_z^{d+1} \subset B(z, \beta) \setminus B(z, \alpha)$. Therefore, letting $M_z = M_z^{d+1}$ concludes the proof. \square

Definition 4.17. Let $S \subset \mathbb{R}^n$ and $\epsilon > 0$. We define S to be ϵ -dense if for any $z \in S$, there exists $z' \in S$ and $z' \neq z$ such that $\|z - z'\| \leq \epsilon$. Let $d \in \mathbb{N}$. We say that S is $d - \epsilon$ -dense if for any $z \in S$, there exists $M \subset S \setminus z$ with $|M| = d$ such that for any $z' \in M$, we have $\|z - z'\| < \epsilon$.

Corollary 4.18. Let $M \subset \mathbb{R}^n$ with $2 \leq a \leq |M|$ where a is the batch size used for training. If M satisfies that for any $S \subset M$ with $|S| = a$, we have S is $\alpha - \beta$ -connected, then it follows that M is $(|M| - a + 1) - \beta$ -dense.

Proof. By Lemma 4.16, for every $z \in M$, we can construct $M_z \subset B(z, \beta) \setminus B(z, \alpha)$ with $|M_z| = |M| - a + 1$. Therefore for any $z' \in M_z$, we have $\|z - z'\| \leq \beta$ \square

Remark 4.19. Lemma 4.16 and Corollary 4.18 give us information on a densification phenomenon for samples with sizes larger than a . In particular, if the sample is mapped to M through the trained f_θ , then there are at least $|M| - a + 1$ latent representations within a distance of β for any $z \in M$. The density around z increases with $|M|$ if a is fixed and increases as a decreases if $|M|$ is fixed.

Now, we will utilize the densification conclusion drawn from Lemma 4.16 and Corollary 4.18 to study further the effect on the size of the sample to a specific structural feature of the latent space which will be introduced in the definition below. This feature will be crucial in explaining the utility of this connectivity loss function to the autoencoder and the task of one-class learning.

Definitions 4.20. Let $S \subset \mathbb{R}^n$ and $\mu > 0$. We say that S is μ -separated if every pair of distinct elements $z, z' \in S$ satisfies $\|z - z'\| \geq \mu$.

For some $X \subset \mathbb{R}^n$, we define the μ -metric entropy of X to be

$$N_\mu(X) = \max\{|S| : S \subset X \text{ and } S \text{ is } \mu\text{-separated}\}$$

Now, our goal is to show that if the sample has enough elements, the sample is not μ -separated for some $\mu > 0$. We will make this statement precise starting with the following corollary.

Corollary 4.21. Let $M \subset \mathbb{R}^n$ with $2 \leq a \leq |M|$ where a is the batch size used for training. If M satisfies that for any $S \subset M$ with $|S| = a$, we have S is $\alpha - \beta$ -connected, then for any $\mu > 0$, if $|M| - a + 1 > N_\mu(B(x, \beta) \setminus B(x, \alpha))$ for any $x \in \mathbb{R}^n$, we have that M is not μ -separated.

Proof. By Lemma 4.16, for any $z \in M$, we can specify $M_z \subset B(z, \beta) \setminus B(z, \alpha)$ with $|M_z| = |M| - a + 1$. Then if $|M| - a + 1 > N_\mu(B(z, \beta) \setminus B(z, \alpha))$, we have that $|M_z|$ is bigger than the largest order of a set that in $B(z, \beta) \setminus B(z, \alpha)$ that can be μ -separated. It follows that M_z is not μ -separated. Because $M_z \subset M$, it follows that M is not μ -separated. \square

Remark 4.22. It should be clear that $N_\mu(B(x, \beta) \setminus B(x, \alpha))$ is invariant across any $x \in \mathbb{R}^n$

Finally, we will try to understand more about $N_\mu(B(x, \beta) \setminus B(x, \alpha))$ and how it's linked to the dimension n of the latent space.

Proposition 4.23. *Let $\mu < 2\alpha$ and $\alpha < \beta$. Under the metric space $(\mathbb{R}^n, \|\cdot\|)$, we have that*

$$N_\mu(B(x, \beta) \setminus B(x, \alpha)) \leq \left(\frac{2\beta}{\mu} + 1\right)^n - \left(\frac{2\alpha}{\mu} - 1\right)^n$$

Proof. We let M be any $M \subset B(x, \beta) \setminus B(x, \alpha)$ for some $x \in \mathbb{R}^n$ and M is μ -separated. Then fix $z \in M$, for any $y \in B(z, \frac{\mu}{2})$ we have $\|y\| \leq \|y-z\| + \|z\| < \frac{\mu}{2} + \beta$ and $\|y\| \geq \|z\| - \|z-y\| \geq \alpha - \frac{\mu}{2}$. It follows that for every $z \in M$, we have $B(z, \frac{\mu}{2}) \subset B(x, \beta + \frac{\mu}{2}) \setminus B(x, \alpha - \frac{\mu}{2})$. Moreover, as M is μ -separated, for any $z, z' \in M$ and $z \neq z'$, $\|z - z'\| \geq \mu \Rightarrow B(z, \frac{\mu}{2}) \cap B(z', \frac{\mu}{2}) = \emptyset$. In conclusion, all $B(z, \frac{\mu}{2})$ with $z \in M$ are disjointly contained in $B(x, \beta + \frac{\mu}{2}) \setminus B(x, \alpha - \frac{\mu}{2})$.

Now, let λ be the Lebesgue measure in \mathbb{R}^n . We then have

$$|M| \cdot \lambda(B(x, \frac{\mu}{2})) = \lambda\left(\bigcup_{z \in M} B(z, \frac{\mu}{2})\right)$$

This is because the Lebesgue measure has the property of countable additivity:

$$\lambda\left(\bigcup_{z \in M} B(z, \frac{\mu}{2})\right) = \sum_{z \in M} \lambda(B(z, \frac{\mu}{2})) = |M| \cdot \lambda(B(x, \frac{\mu}{2}))$$

As all $B(z, \frac{\mu}{2})$ with $z \in M$ are contained in $B(x, \beta + \frac{\mu}{2}) \setminus B(x, \alpha - \frac{\mu}{2})$ and λ is translation invariant, we consequently have $\lambda\left(\bigcup_{z \in M} B(z, \frac{\mu}{2})\right) \leq \lambda(B(x, \beta + \frac{\mu}{2}) \setminus B(x, \alpha - \frac{\mu}{2}))$,

which gives us

$$|M| \cdot \lambda(B(x, \frac{\mu}{2})) \leq \lambda(B(x, \beta + \frac{\mu}{2}) \setminus B(x, \alpha - \frac{\mu}{2}))$$

Given that $\lambda(B(x, r)) = \frac{2^n}{n!} r^n$, we have the volumes of balls in the above equation with this formula and get

$$\begin{aligned} |M| \cdot \frac{\mu^n}{n!} &\leq \frac{2^n}{n!} \left(\left(\beta + \frac{\mu}{2}\right)^n - \left(\alpha - \frac{\mu}{2}\right)^n \right) \\ |M| &\leq \frac{2^n}{\mu^n} \left(\left(\beta + \frac{\mu}{2}\right)^n - \left(\alpha - \frac{\mu}{2}\right)^n \right) \\ &= \left(\frac{2\beta}{\mu} + 1\right)^n - \left(\frac{2\alpha}{\mu} - 1\right)^n \end{aligned}$$

As M is an arbitrary μ -separated set in $B(x, \beta) \setminus B(x, \alpha)$, we conclude that $N_\mu(B(x, \beta) \setminus B(x, \alpha)) \leq \left(\frac{2\beta}{\mu} + 1\right)^n - \left(\frac{2\alpha}{\mu} - 1\right)^n$. \square

Finally, [Corollary 4.21](#) and [Proposition 4.23](#) together provide a way to assess whether a sample M has enough elements to be not μ -separated, which is stated in the following corollary.

Corollary 4.24. *Let $M \subset \mathbb{R}^n$ with $2 \leq a \leq |M|$ where a is the batch size used for training. If M satisfies that for any $S \subset M$ with $|S| = a$, we have S is $\alpha - \beta$ -connected, then if $|M| - a + 1 > \left(\frac{2\beta}{\mu} + 1\right)^n - \left(\frac{2\alpha}{\mu} - 1\right)^n$, it follows that M is not μ -separated*

In particular, if we let $\mu = \eta$ which is the value of η specified for the connectivity loss as defined in [Definition 4.9](#), we have that if $|M| - a + 1 > (\frac{2\beta}{\eta} + 1)^n - (\frac{2\alpha}{\eta} - 1)^n$, then M is not η -separated.

4.2.4. Connectivity Loss and One-Class Learning. Finally, as promised, we will now see briefly how is this linked to the task of one-class learning.

The task of one-class learning is stated as the following: let $C \subset X$ be a class of data. We are given m training data points all coming from C , i.e. $T = \{x_1, x_2, \dots, x_m\} \subset C$. We want to be able to determine whether any given sample $y \in X$ belongs to class C .

In a nutshell, we train the autoencoder with the reconstruction loss in combination with the connectivity loss. After that, we will ideally have that $M = z_1, z_2, \dots, z_m = f_\theta(T)$ satisfies that for any $S \subset M$ with $|S| = a$ where a is the batch size in training, we have S is $\alpha - \beta$ -connected for some $\alpha < \beta$ as mentioned in [Section 4.2.3](#).

Now we calculate the latent representation of y and denote it z_y . If y belongs to C , given that the autoencoder captures the feature of class C by its reconstruction function into the connectivity structure on the latent space with the assistance of the connectivity loss. Ideally, the addition of z_y into M should preserve the connectivity structure and any $S \subset M \cup \{z_y\}$ with $|S| = a$ still imply that S is $\alpha - \beta$ -connected.

Therefore, as long as $|T|$ is large enough, for any $M' \subset M \cup z_y$ that satisfies $|M'| - a + 1 > (\frac{2\beta}{\eta} + 1)^n - (\frac{2\alpha}{\eta} - 1)^n$, M' is not η -separated, i.e. there exists $z \in M'$ that has a neighbor within the distance of η according to [Corollary 4.24](#). With $|T|$ large, there will be lots of M' which contains z_y that satisfies this condition. This leads to a large value of the following:

$$s(y) = |\{z_i : \|z_i - z_y\| \leq \eta, 1 \leq i \leq m\}|$$

Therefore, we can take $s(y)$ as a score for the membership of a given point $y \in X$ to be in class C . This score function requires no optimization but only the connectivity structure of the latent space. At this stage, we can see how such a structural loss function can help with the task of one-class learning for an autoencoder. It enables us to use the structure of the latent space itself to efficiently evaluate the membership of a given point in a class.

ACKNOWLEDGMENTS

I'd like to express my gratitude to my mentor, Becky Zhang, for her invaluable guidance throughout the topic selection, learning process, and paper writing. Her dedicated support in teaching me algebraic topology from scratch, with regular, active communication and access to essential materials, has been invaluable.

I'm also thankful to Professor Peter May for organizing the exceptional Math REU program. This program has allowed me to achieve more than I ever thought possible within a short timeframe. Special thanks to Professors Danni Rudenko and Laszlo Babai for their excellent mentorship.

Lastly, I want to extend my heartfelt appreciation to my family and friends for their unwavering support, without which I wouldn't have been able to complete this program or anything else.

REFERENCES

- [1] Edelsbrunner, et al. “Topological Persistence and Simplification.” *Discrete & Computational Geometry*, vol. 28, no. 4, 2002, pp. 511–533, doi:10.1007/s00454-002-2885-2.
- [2] EDELSBRUNNER, HERBERT. *Computational Topology: An Introduction*. AMER MATHEMATICAL SOCIETY, 2022.
- [3] Hofer, C., Kwitt, R., Niethammer, M. & Dixit, M.. (2019). Connectivity-Optimized Representation Learning via Persistent Homology. *Proceedings of Machine Learning Research*97:2751-2760 Available from <https://proceedings.mlr.press/v97/hofer19a.html>.
- [4] Baldi, Pierre. *Deep Learning in Science*. Cambridge University Press, 2021.
- [5] Chapter 4: Persistent Homology *Topics in Computational Topology: An ...*, web.cse.ohio-state.edu/~wang.1016/courses/5559/Lecs/Chap4-Persistence.pdf.
- [6] MA 331: Topology, personal.colby.edu/~sataylor/teaching/F14/MA331/MA331.php.